AMENDMENTS TO THE CLAIMS:

This listing of claims will replace all prior versions, and listings, of claims in the application:

LISTING OF CLAIMS:

1-21. (cancelled)

22. (previously presented) A multiprocessor system comprising:

a plurality of processors which send and receive predetermined information to and from each other;

a shared memory which is shared and accessed by each of said plurality of processors; and

an access manager which manages access to said shared memory by each of said plurality of processors,

wherein, when said plurality of processors are in contention to access said shared memory, said access manager selects one of said plurality of processors and permits said one of said plurality of processors to access said shared memory,

wherein, once each of said plurality of processors has accessed said shared memory, when said one of said plurality of processors updates a predetermined data in said shared memory, said one of said plurality of processors requests others of said plurality of processors to access said updated predetermined data from said shared memory.

23. (previously presented) The multiprocessor system according to claim 22, wherein said one of said plurality of processors requests others of said plurality of processors to update said updated predetermined data from said shared memory.

24. (previously presented) The multiprocessor system according to claim 22, wherein said one of said plurality of processors requests others of said plurality of processors to update the predetermined data in said shared memory, when said others of said plurality of processors have not yet updated said predetermined data.

25. (previously presented) The multiprocessor system according to claim 22, wherein, when a predetermined period of time has elapsed without being selected by said access manager, said one of said plurality of processors requests others of said plurality of processors and said access manager to perform a predetermined reset operation for resetting themselves.

26. (previously presented) A multiprocessor system comprising:

a plurality of processors which send and receive a predetermined signal to and from each other;

a shared memory which is shared and accessed by each of said plurality of processors; and

a contention determiner which detects whether said plurality of processors are in contention to access said shared

memory, and permits one of said plurality of processors to access said shared memory,

wherein, once said one of said plurality of processors has accessed said shared memory said one of said plurality of processors outputs an access-request signal to another one of said plurality of processors, so as to request said another one of said plurality of processors to access said shared memory, and

wherein once each of said plurality of processors has accessed said shared memory, when said one of said plurality of processors updates a predetermined data in said shared memory, said one of said plurality of processors outputs a re-read request signal to another of said plurality of processors, so as to request said others of said plurality of processors to access said updated predetermined data from said shared memory.

27. (currently amended) The multiprocessor system according to claim 26, wherein said one of said plurality of processors outputs an update-request signal, so as to request said ~~other~~ others of said plurality of processors to update said updated predetermined data from said shared memory.

28. (previously presented) The multiprocessor system according to claim 26, wherein said one of said plurality of processors outputs an update-request signal, so as to request said others of said plurality of processors to update the predetermined data in said shared memory, when said others of

said plurality of processors have not yet updated said predetermined data.

29. (previously presented) The multiprocessor system according to claim 26, wherein, when a predetermined period of time has elapsed without being selected by said access manager, said one of said plurality of processors outputs a reset-request signal to said others of said plurality of processors and said contention determiner, so as to request said others of said plurality of processors and said contention determiner to reset themselves.

30. (previously presented) A shared-memory controlling method to be executed in a multiprocessor system including a plurality of processors which send and receive predetermined information to and from each other, a shared memory which is shared and accessed by each of said plurality of processors, and an access manager which manages access to said shared memory by each of said plurality of processors, said method comprising:

selecting one processor of said plurality of processors, and permitting said one processor to access said shared memory, when said plurality of processors are in contention for said shared memory;

performing a first access to said shared memory using said one processor;

requesting others of said plurality of processors to perform a second access to said shared memory, when said performing the first access to said shared memory has been done; and

performing the second access to said shared memory using said others of said plurality of processors,

wherein, once each of said plurality of processors has accessed said shared memory, said one processor updates a predetermined data in said shared memory and requests said others of said plurality of processors to access said updated predetermined data from said shared memory.

31. (previously presented) The shared-memory controlling method as claimed in claim 30, wherein said requesting step includes said one processor requesting others of said plurality of processors to update said updated predetermined data in said shared memory.

32. (previously presented) The shared-memory controlling method according to claim 30, wherein said requesting step includes said one processor requesting others of said plurality processors to update the predetermined data in said shared memory.

33. (previously presented) The shared-memory controlling method according to claim 30, wherein, when a predetermined period of time has elapsed without being selected

in said selecting step, said requesting step includes requesting said others of said plurality of processors and said access manager to perform a predetermined reset operation for resetting themselves.

34. (previously presented) A shared-memory controlling method comprising:

selecting one processor of a plurality of processors, and permitting said one processor to access a shared memory shared by the plurality of processors, when the plurality of processors are in contention for the shared memory;

performing a first access to said shared memory using said one processor;

requesting others of said plurality of processors to perform a second access to said shared memory, when the first access has been done; and

performing the second access to said shared memory using the others of said plurality of processors;

wherein, once each of said plurality of processors has accessed said shared memory, said one processor updates a predetermined data in said shared memory and requests said others of said plurality of processors to access said updated predetermined data from said shared memory.

35. (previously presented) A computer readable recording medium for controlling a computer to execute a shared-memory controlling method comprising:

selecting one processor included in a plurality of processors, and permitting said one processor to access a shared memory, when said plurality of processors are in contention for said shared memory;

performing a first access to said shared memory using the selected processor;

requesting others of said plurality of processors to perform a second access to said shared memory, when said performing the first access to said shared memory has been done; and

performing the second access to said shared memory using said others of said plurality of processors,

wherein, once each of said plurality of processors has accessed said shared memory, when said one processor updates a predetermined data in said shared memory in performing the first access, said one processor requests said others of said plurality of processors to access said updated predetermined data from said shared memory.

36. (previously presented) A data signal embedded in a carrier wave and representing an instruction sequence for

controlling a computer to execute a shared-memory controlling method comprising:

selecting one processor of a plurality of processors, and permitting said one processor to access a shared memory shared by the plurality of processors, when the plurality of processors are in contention for the shared memory;

performing a first access to said shared memory using said one processor;

requesting others of said plurality of processors to perform a second access to said shared memory, when the first access has been done; and

performing the second access to said shared memory using the others of said plurality of processors;

wherein, once each of said plurality of processors has accessed said shared memory, when said one processor updates a predetermined data in said shared memory in performing the first access, said one processor requests said others of said plurality of processors to access said updated predetermined data from said shared memory.